Transcoding in the Cloud: Optimization and Perspectives

Ramon Aparicio-Pardo, Gwendal Simon, Alberto Blanc

Transcoding in the Cloud: Optimization and Perspectives

Ramon Aparicio-Pardo, Gwendal Simon° and Alberto Blanc°*
** University of Nice, ramon.aparicio-pardo@unice.fr*
°Telecom Bretagne, France, firstname.lastname@telecom-bretagne.eu

1. Introduction

The most popular online video services are now hosted "in the cloud". This now mainstream idiom indicates a set of hardware and software technologies. On the hardware side, the servers in the data-centers offer computing resources for the preparation of the video content, while the *edge-servers* in the Content Delivery Network (CDN) store and deliver the video streams. On the software side, the encoders transform a flow of data depicting images into a compressed, transportable stream. Software also implements the rate-adaptive streaming strategies and is in charge of managing the CDN resources, taking into account various technical and business constraints. The coordination between all software and hardware technologies is a key requirement, affecting both the Quality of Experience (QoE) of the end-users, who consume the video streams, and the operational costs of the service provider.

In this letter, we describe research activities on the global management of a live video streaming service running in the cloud. We focus on the coordination of these technologies and we show that a management policy that takes into account all the inter-dependencies among technologies can bring significant advantages. In particular, we address the problem of preparing the adequate video *package* (the set of representations) taking into account multiple constraints.

We first describe the main elements of a cloud streaming system. We then introduce an optimization framework, which has been presented in more details in [1,2]. This framework can be tuned to reflect the objectives and constraints of different actors. We then present some results of trace-driven performance evaluations to illustrate the different results that can be obtained by the framework. Finally, and most importantly, we discuss the perspectives of the research on cloud optimization from a global standpoint, taking into account both business and scientific concerns.

2. Cloud Streaming Chain

Figure 1 shows a simplified view of the main components, and corresponding actors, of a typical cloud streaming chain. Each one is briefly described in the following.

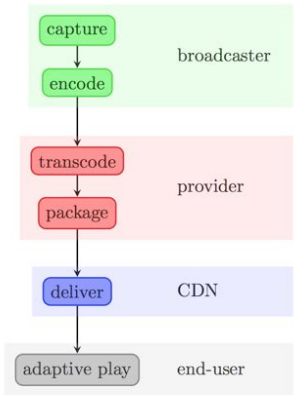

Figure 1: Cloud Streaming Chain

2.1. Broadcaster

The broadcaster is the entity that generates the original video. It can be either a professional broadcaster, such as a traditional TV provider, or, more recently, an individual source of video, such as people broadcasting while playing a videogame [3,4] or while attending all sort of events [5].

In the case of professional broadcaster, the video is captured, and then encoded by so-called *contribution encoders*. The main goal of video contribution solutions is to ensure that the raw video is encoded in a high-quality high-fidelity manner, but at a bit-rate that respects the network condition between the server that host the contribution encoder (usually enclosed or near the camera) and the *entrypoint server* of the video provider. For user-generated live streaming systems, where individuals capture and upload the scene, a new generation of software has been developed, such as Open Broadcaster Software (OBS)[1] and screencasters [5]. The main idea here is to find trade-offs between the quality of the compression and the capabilities of the machine hosting the encoders (typically a smartphone for services like Meerkat and Periscope and a machine that is almost fully utilized for running games in the case of Twitch).

2.2. The Cloud

The term cloud is used to describe all the operations that are done "somewhere" in the Internet, neither at the broadcaster side, nor at the end-user side. We

[1] https://obsproject.com/

distinguish the roles of video providers and CDNs, the former being the main provider of the service while the latter is an intermediary, but key, actor. Some video companies play both roles but this is not always the case.

The video provider gets as input the "original video" (more precisely the video that is the output of the contribution encoders). Its role is to prepare the video so that end-users will be able to eventually play it on their device. In the recent years, the heterogeneity of end-users' device has grown significantly, from smartphones to connected TVs, and even new generations of Virtual Reality (VR) headsets. To address this heterogeneous population of end-users, the video providers have adopted rate-adaptive streaming systems such as Apple's HTTP Live Streaming (HLS) and the MPEG standard Dynamic Adaptive Streaming over HTTP (DASH).

The implementation of rate-adaptive streaming first requires a transcoding operation. For each video, the video provider should generate k different *representations*, each of them characterized by a different bit-rate, resolution, sometimes frame rate and key frame period. Then, the video provider should package the video by aggregating the set of representations, by segmenting the representations and by creating the manifest file, which is the file that describes the playlist.

The cost of transcoding these videos can strain the computing infrastructure of video providers. Typically in Twitch, more than 6,000 videos need to be transcoded in average [3]. Furthermore video providers increase the number of representations per video so that each end-user can find a good match among the available representations. To meet this demand, the video providers manage large data-centers with thousands of servers [8].

The CDN gets in input the package of videos that has been prepared by the video provider. Its role is to deliver the content. The literature related to live streaming in CDN provides details about the processes that are implemented in CDN to make sure that an end-user that requests a given segment of a given representation can find a nearby edge-server that stores the said segment [6,7].

The number of edge-servers in a CDN (more than 200,000 for the biggest CDN players) is expected to enable large-scale delivery in good conditions. However, the costs of transporting the packages of video representations from the CDN's origin server to the subset of edge-servers that must deliver this content have grown. Indeed, the set of representations includes many Ultra-High-Definition (UHD) and High-Definition (HD) videos, which have a large bit-rate. CDNs need to reserve a large bandwidth in the core network to deliver the segments to the edge servers.

2.3. The End-Users

The endpoint of the chain is the video player of the end-user. With the adoption of adaptive streaming, the role of the video player is no longer passive; on the contrary, it is responsible for selecting the right representation for every segment of the video. Various strategies have been devised to efficiently choose the representation whose bit-rate is closest to the currently available network capacity, while avoiding to change representation too often. Recent studies have highlighted the relations between the engagement of users in a service and the QoE of the video streaming [9]. These studies have also revealed that many parameters impact the QoE: of course the video bit-rate, but also the delay to start the video playback, and most importantly the interruptions due to video re-buffering. Another aspect that has not been extensively studied, to the best of our knowledge, is the relation between the QoE and the device that is used to play the video. The size of the display screen is typically one of the parameters that the video provider should also consider for the transcoding of the representations.

3. Optimization Models

We briefly introduce in this Section two models that we have described in [1,2]. Both models aim at helping the video provider when deciding which representations to generate.

3.1. Video Type, Popularity and CDN Budget

To the best of our knowledge, the first paper to study encoding choices for adaptive streaming is [1]. The main idea is that every ingested video coming from the contribution encoders can be transcoded as many times as necessary. However, the transcoder impacts all the following modules in the delivery chain. In particular, the more video representations are created, the larger the CDN budget to transport the full package to the edge-servers. Moreover, the load on the packager also depends on the number of representations. To simplify the management and the operational cost of the video service, it is thus preferable to limit the overall number of representations K.

In the industry, the norm is to transcode every video into the same number of representations with the same encoding parameters. In [1], we show that this solution is far from optimal. Instead, it is better to decide for each video the number of representations to transcode and the profile of these representations. In particular, we emphasize the benefits that one can expect from following some intuitive rules related to the popularity of the video, the nature of the video, and the devices of the target population.

To illustrate our claim, we first showed that the recommendations for transcoding given by the main

video players are sub-optimal. To validate this claim, we used an Integer Linear Program (ILP) to compute the optimal set of representations for any given overall number of representations *K*.

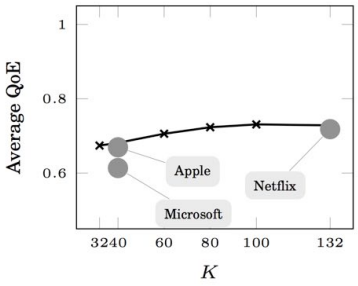

Figure 2: Average QoE for a given overall number of representations K [1]

Figure 2 shows the average QoE of 500 users watching four different videos. The circles correspond to the recommendations by Microsoft, Apple and Netflix and the line with the crosses to the optimal solution obtained with the ILP." This figure shows not only that the representation recommended by Apple and Microsoft are sub-optimal and use too few representations but also that with half the representations recommended by Netflix it is possible to almost achieve the same QoE.
More details about simulations settings, as well as many other simulation results can be found in [1].

3.2. Transcoders in Data-Centers

In [2], we take into account the processing power that is required to transcode each given video stream in real time. If we refer to the delivery chain of Figure 1, we considered the packaging load and the CDN budget as the two main constraints in [1], while in [2] we add the transcoding load as an additional constraint. This latter constraint is likely to be the preponderant one in many cases, and in particular for services like Twitch, where a large number of video streams have to be transcoded.
To study this problem, we collected two large datasets, which are also publicly available. The first one is the result of a of four month study of Twitch [3]. We highlight two main characteristics of the ingested streams: the number of streams can vary significantly within a day, and the diversity of the quality of ingested video is large. Our second dataset is a comprehensive set of measurements that we realized on a series of systematic transcoding of various videos, from any resolution and bit-rate to any other resolution and bit-rate. We focus on the number of CPU cycles that are required to make these transcoding operations and the quality of the transcoded videos compared to the quality of the original videos.
The main claim of [2] is similar as the one in [1], that is, the video provider can obtain better performance if all videos are not transcoded in the same way. We used an ILP to find the optimal number of representations and the corresponding parameters (e.g., resolution, bit-rate) taking into account constraints on the number of CPU cycles available. Figure 3 shows the result of our tests. Again, the optimal set of packages provides a far better quality when compared to the standard choices in the market.

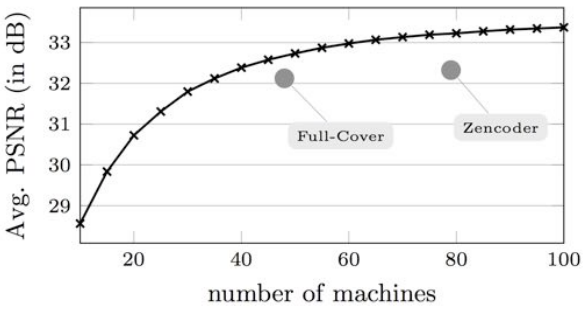

Figure 3: Quality of the videos as a function of the number of machines used.

We then propose and implement a heuristic, which is not optimal but can be easily and efficiently implemented, to decide the number of representations and their profiles, for every ingested stream. We apply this heuristic to various snapshots of the Twitch system, and compare both the number of CPU cycles that the transcoding operations consumed in the datacenter and the average quality of the videos with respect to the original video. We show the result in Figure 4.

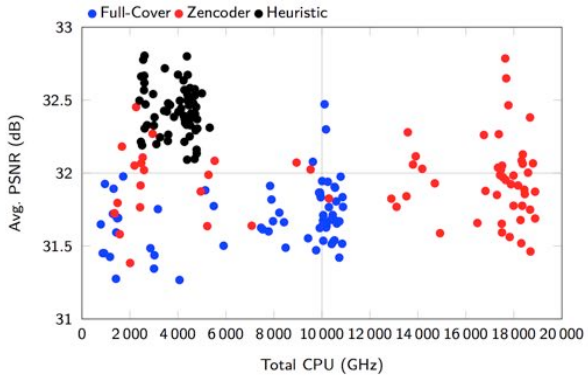

Figure 4: The average quality and the number of consumed CPU cycles for various snapshots of the Twitch system

We especially emphasize in Figure 4 that the traditional approaches cannot accommodate variable load in inputs. The Zencoder solution requires sometimes the reservation of a large-scale datacenter while it requires a small one at other moments of the day. On the contrary a smart management can make sure that a given amount of resources are always fully exploited.

4. Discussion

The studies in [1,2] have shown that significant gains can be obtained by implementing smart strategies for the transcoding of the videos in the context of adaptive streaming. However, to turn our solutions into practical implementations, the actors involved in the delivery chain have to share more information than they currently do. In particular, the set of servers in the datacenter, the packager load, the overall charge on the CDN and the characteristics of the population of users (or at least the size of the population consuming a given video in real-time) are key information that are needed at several stages in the chain. Today, the video provider and the CDN are not tightly coupled (with the exception of vertical integrated companies like Google and Netflix, which manage all combined video services, datacenters and CDNs).

Unsurprisingly, two leading companies specialized in video transcoding in datacenters (namely Envivio and Elemental) have both been recently acquired by larger companies (Ericsson and Amazon respectively). Such events prove that the vertical integration of a maximum number of key actors in the delivery chain is seen as a way to both generate significant savings in operational costs, but also to improve the performance of the video services. Important optimization processes can be put in place when one has a global view of the system.

In the near future, the collaboration between the CDN and the datacenter in charge of transcoding the video should be further improved. The transcoder needs information from the CDN about the end-users who consume the videos and about the network conditions that the CDN has to deal with. On its side, the CDN can optimize the delivery and significantly reduce the operational costs if it knows the underlying structure of the streams that it should transport from the origin server to the edge-servers. All this calls for a better collaboration, possibly by the mean of standard APIs, between these actors.

References

[1] Laura Toni, Ramon Aparicio, Alberto Blanc, Gwendal Simon, and Pascal Frossard. "*Optimal Set of Video Representations in Adaptive Streaming*", in ACM MMSys, 2014.

[2] Ramon Aparicio, Karine Pires, Alberto Blanc and Gwendal Simon. "*Transcoding Live Video Streams at a Massive Scale in the Cloud*" in ACM MMSys, 2015.

[3] Karine Pires and Gwendal Simon. "*DASH in Twitch: Adaptive Bitrate Streaming in Live Game Streaming Platforms*" in ACM VideoNext Conext Workshop, 2014.

[4] Ryan Shea, Di Fu, and Jiangchuan Liu "*Towards bridging online game playing and live broadcasting: design and optimization*", in ACM Nossdav, 2015.

[5] Chih-Fan Hsu, Tsung-Han Tsai, Chun-Ying Huang, Cheng-Hsin Hsu, and Kuan-Ta Chen. "*Screencast dissected: performance measurements and design considerations*", in ACM MMSys, 2015.

[6] Matthew Mukerjee, David Naylor, Junchen Jiang, Dongsu Han, Srinivasan Seshan, Hui Zhang. "*Practical, Real-time Centralized Control for CDN-based Live Video Delivery*", in ACM Sigcomm, 2015

[7] Jiayi Liu, Gwendal Simon, Géraldine Texier, and Catherine Rosenberg. "*User-centric discretized delivery of rate-adaptive live streams in underprovisioned CDN networks*", IEEE Journal in Selected Areas in Communications, 2014.

[8] Luiz A. Barroso, Jimmy Clidaras, and Urs Hölzle. "*The datacenter as a Computer: An Introduction to the Design of Warehouse-scale Machines*". Morgan Claypool, 2013.

[9] S. Shunmuga Krishnan and Ramesh Sitaraman "*Video Stream Quality Impacts Viewer Behavior: Inferring Causality using Quasi-Experimental Designs*". in ACM Internet Measurement Conference (IMC), 2012.

Ramon Aparicio-Pardo received a Ph.D. Degree in Information and Communication Technologies from Universidad Politécnica de Cartagena (UPCT), Spain, in 2011. After that, he completed a postdoctoral fellowship with Orange Labs in 2012. He has then worked as a postdoc researcher at Telecom Bretagne from 2013 to 2015. He is now Associate Professor at University of Nice. His research interests include planning, design and evaluation of communication networks by means of mathematical optimization.

Gwendal Simon graduated from University Rennes (France). During his PhD, he worked at Orange Research Labs. From 2004 to 2006 he was a researcher at Orange Labs. Since 2006, he has been Associate Professor at Telecom Bretagne, a graduate engineering school within the Institut Mines-Telecom. He was a visiting researcher at University of Waterloo from September 2011 to September 2012. His research interests include multimedia delivery systems (video and gaming) and network management.

Alberto Blanc is an associate professor at Telecom Bretagne since 2010. He received a "laurea" in computer engineering in 1998 from the "Politecnico di Torino," Italy, and a Ph.D. in electrical engineering from the University of California, San Diego, U.S., in 2006. His research interests include performance evaluation of computer networks and cloud computing.